IN THE CLAIMS:

Please cancel claim 15.

Please amend the claims as follows:

1. (~~Once~~ Twice Amended) In a data processing system including one or more requesting processors that generate processor requests directed to one or more peripheral devices, [A] a [multiple priority] non-blocking load buffer, comprising:

a plurality of variable depth pending queues corresponding to each one of the plurality of peripheral devices for queuing entries of [memory or I/O requests generated by] processor requests[s to peripheral devices, each of said pending queues including a plurality of sub-queues; and];

a variable length return queuing unit for [buffering] queuing data returned from said peripheral devices in response to [said I/O] an outstanding processor request; and

a free pool of entries for placing entries of the outstanding processor request, after the outstanding processor request is accepted by a corresponding ~~requesting processor~~ peripheral device.

2. (Once Amended) A [multiple priority] non-blocking load buffer according to Claim 1, wherein said variable length return queuing unit comprises one variable length return queue.

27

3. (Once Amended) A [multiple priority] non-blocking load buffer according to Claim 1, wherein said variable length return queuing unit comprises a plurality of variable length return queues.

4. (Once Amended) A [multiple priority] non-blocking load buffer according to Claim 3, wherein each of said plurality of variable length return queues corresponds to each of said requesting processors.

B7

5. (Once Amended) A [multiple priority] non-blocking load buffer according to Claim 3, wherein each of said plurality of variable length return queues corresponds to each of said peripheral devices.

6. (Once Amended) A [multiple priority] non-blocking load buffer according to Claim 3, wherein each of said plurality of variable length return queues corresponds to a unique priority level.

7. (Once Amended) A [multiple priority] non-blocking load buffer according to Claim 1, wherein each of said pending queues include a plurality of sub-queues [within one of said pending queues], wherein each of said sub-queues is assigned a unique priority level.

28

8. (Once Amended) A [multiple priority] non-blocking load buffer according to Claim 7, wherein said [I/O] processor requests include an address and a priority tag, said address directs said [memory or I/O] processor requests to a corresponding one of said pending queues and said priority tag channels said [memory or I/O] processor requests to a corresponding one of said sub-queues within the one said pending queue.

9. (Once Amended) A [multiple priority] non-blocking load buffer according to Claim 8, wherein each of said pending queues comprises a priority controller for issuing [said memory or I/O] the processor requests from said sub-queues in a highest priority first manner.

B2 end

10. (Once Amended) A multiple priority non-blocking load buffer comprising:

a variable depth pending queue for queuing entries of memory or I/O requests generated by a processor to peripheral devices, said pending queue including a plurality of sub-queues with each sub-queue having a unique priority level assigned thereto; [and]

a variable length return queuing unit for [buffering] queuing data returned from said peripheral devices in response to [said] an outstanding memory or I/O request; and

a free pool of entries for placing entries of the outstanding I/O request, after the outstanding I/O request is accepted.

15 ~~16.~~ (Once Amended) A multiple priority non-blocking load buffer according to Claim [15]10, wherein said memory or I/O requests include a priority tag, said priority tag channels said memory or I/O requests to a corresponding one of said sub-queues.

B3

17
~~21.~~ (Once Amended) A [multiple priority] non-blocking load buffer according to Claim 7, wherein a maximum number of outstanding [memory or I/O transactions] processor requests is specified for said unique priority level in each of said sub-queues which prevents entries of [memory or I/O] processor requests having low priority levels from using one of said sub-queues before entries of [memory or I/O] processor requests having higher priority levels.

B4

18
~~22.~~ (Once Amended) A [multiple priority] non-blocking load buffer according to Claim 1, wherein priorities corresponding to the entries of [memory or I/O] processor requests are determined by logical memory addresses, control bits derived from a memory management page table, control bits derived from segmentation entries, virtual addresses of a memory management system, programmable registers which set priorities for each processor, instructions, or instruction operand.

19
~~23.~~ (Once Amended) A multiple priority non-blocking load buffer according to Claim [15]10, wherein a maximum number of outstanding memory or I/O [transactions] requests is specified for said unique priority level in each of said sub-queues which

prevents entries of memory or I/O requests having low priority levels from using one of said sub-queues before entries of memory or I/O requests having higher priority levels.

20
~~24~~. (Once Amended) A multiple priority non-blocking load buffer according to Claim 10, wherein priorities corresponding to entries of memory or [I/O] requests are determined by logical memory addresses, control bits derived from a memory management page table, control bits derived from segmentation entries, virtual addresses of a memory management system, programmable registers which set priorities for each processor, instructions, or instruction operand.

Please add the following new claims:

27. (New claim) A non-blocking load buffer for a multi-processor system running real-time processes, comprising:

a memory array for storing data; and

a control block for simultaneously performing a plurality of memory or I/O transactions.

28. (New claim) A non-blocking load buffer according to Claim 27, wherein said plurality of memory or I/O transactions comprise loads.

29. (New claim) A non-blocking load buffer according to Claim 27, wherein said plurality of memory or I/O transactions comprise stores.

31

30. (New claim) A non-blocking load buffer according to Claim 27, wherein control block comprises a plurality of queues for temporarily storing data.

31. (New claim) A non-blocking load buffer according to Claim 30, wherein said control block comprises pointers to entries in said memory array are queued.

32. (New claim) A non-blocking load buffer according to Claim 30, wherein a plurality of independent pending queues correspond to each of a plurality of peripherals connected to the non-blocking load buffer.

33. (New claim) A non-blocking load buffer according to Claim 27, wherein the data stored in said memory array comprises a plurality of entries each including address, data and control information.

34. (New claim) A non-blocking load buffer according to Claim 27, wherein said memory array is divided into a plurality of portions including address/control memory array portions and data memory array portions.

21

~~35.~~ (New claim) A non-blocking load buffer according to claim 1, wherein the variable depth pending queues queue the entries of processor requests using the free pool of entries.

32

22
~~36~~. (New claim) The data processing system according to claim 1, wherein the non-blocking processor requests are generated by running real-time processes.

23
~~37~~. (New claim) The data processing system according to claim 1, wherein the processor requests are prioritized in the pending queues such that the higher priority processor requests are processed before the lower priority processor requests.

24 23
~~38~~. (New claim) The data processing system according to claim ~~37~~, wherein the pending queues are prioritized such that the higher priority pending queues have a higher number of maximum entries than the lower priority pending queues.

25
~~39~~. (New claim) The data processing system according to claim 1, wherein the one or more requesting processors generate the processor requests over a first bus, and wherein the peripheral devices accept the processor requests over a second bus that has a different bus bandwidth from the first bus.

26 25
~~40~~. (New claim) The data processing system according to claim ~~39~~, wherein the entries include pointers that point to memory locations on a shared memory device.

27 26
~~41~~. (New claim) The data processing system according to claim ~~40~~, wherein the processor requests include control information, addresses and data, and wherein the